L0-norm Constrained Autoencoders for Unsupervised Outlier Detection

Yoshinao Ishii(✉), Satoshi Koide, and Keiichiro Hayakawa

Toyota Central R&D Labs., Inc., Nagakute, Aichi 480–1192, Japan
{y-ishii,koide,kei-hayakawa}@mosk.tytlabs.co.jp

Abstract. Unsupervised outlier detection is commonly performed using reconstruction-based methods such as Principal Component Analysis. A recent problem in this field is the learning of low-dimensional nonlinear manifolds under L0-norm constraints for error terms. Despite significant efforts, no method that consistently treats such features exists. We propose a novel unsupervised outlier detection method, L0-norm Constrained Autoencoders (L0-AE), based on an autoencoder-based detector with L0-norm constraints for error terms. Unlike existing methods, the proposed optimization procedure of L0-AE provably guarantees the convergence of the objective function under a mild condition, while neither the relaxation of the L0-norm constraint nor the linearity of the latent manifold is enforced. Experimental results show that the proposed L0-AE is more robust and accurate than other reconstruction-based methods, as well as conventional methods such as Isolation Forest.

1 Introduction

Unsupervised outlier detection has attracted much attention because it does not require time-consuming manual annotation. Reconstruction-based methods, such as Robust Principal Component Analysis (RPCA), are popular approaches for unsupervised outlier detection [5,12]. A recent trend is the use of nonlinear models, particularly neural network models [1,20,24]. For example, the Robust Deep Autoencoder (RDA) [24] learns a low-dimensional nonlinear manifold where normal samples are located using an autoencoder (AE) [10]. These reconstruction-based methods assume that the feature vector of each sample may include outlier elements; therefore, it is necessary to learn low-dimensional nonlinear manifolds to avoid the impact of outliers.

For robustness to outliers, the l_0-norm is often used for optimization; however, due to its combinatorial property, the optimization is difficult [4]. To avoid such difficulty, relaxation methods, i.e., the use of the l_1-norm or other convex regularization terms, are used. This is, however, problematic because the learned low-dimensional nonlinear manifold is affected by the values of outlier elements, especially in corrupted data. We describe these methods in detail in Sect. 2.

In this paper, we propose L0-norm Constrained Autoencoders (L0-AE), a novel reconstruction-based unsupervised outlier detection method that can learn low-dimensional manifolds under an l_0-norm constraint for the error term using AE.

H. W. Lauw et al. (Eds.): PAKDD 2020, LNAI 12085, pp. 674–687, 2020.
https://doi.org/10.1007/978-3-030-47436-2_51

Table 1. Comparison of features of reconstruction-based methods

	PCA	RPCA	AE	RDA	L0-AE
Decomposition	$X = L + E$	$X = L + S$	$X = \bar{L} + E$	$X = L_D + S$	$X = \bar{L} + S + E$
Minimization	$\|\|X - L\|\|_2^2$	$\|\|L\|\|_* + \lambda\|\|S\|\|_1$	$\|\|X - f_{AE}(X;\theta)\|\|_2^2$	$\|\|L_D - f_{AE}(L_D;\theta)\|\|_2 + \lambda\|\|S\|\|_1$	$\|\|X - f_{AE}(X;\theta) - S\|\|_2^2$
Constraints	$\text{rank}(L) \leq k$	$\|\|X - L - S\|\|_2^2 = 0$	-	$X - L_D - S = 0$	$\|\|S\|\|_0 \leq k$
Convexity	Yes	Yes	No	No	No
Nonlinear Model?	No	No	Yes	Yes	Yes
Considering l_0-norm	No	Yes	No	No	Yes
Convergence of Alternating Optim.	-	-	-	Not proved	Guaranteed if AE is trained appropriately

Table 1 compares the features of different reconstruction-based methods. Compared with the other reconstruction-based methods, L0-AE can provably guarantee the convergence of optimization under the l_0-norm constraint and treat nonlinear features. The key contributions of this work are as follows:

1. We propose a new alternating optimization algorithm that can decompose data nonlinearly under an l_0-norm constraint for the error term (Sect. 3.1).
2. We prove that our alternating optimization algorithm converges under a mild condition, which demonstrates the stability of our algorithm (Sect. 3.2).
3. Through extensive experiments, we show that L0-AE achieves not only high detection accuracy but also stable convergence properties (Sect. 5).

2 Preliminaries

In this section, we describe related reconstruction-based methods. Throughout the paper, we denote a given data matrix by $X \in \mathbb{R}^{N \times D}$, where N and D denote the number of samples and feature dimensions of X, respectively.

Robust PCA: RPCA [5], a robustified version of PCA [12], decomposes X into a low-rank matrix L and a sparse error matrix S such that $X = L + S$ by solving the optimization problem

$$\min_{L,S} \text{rank}(L) + \lambda||S||_0 \quad \text{s.t. } ||X - L - S||_2^2 = 0, \tag{1}$$

where $||\cdot||_0$ is the l_0-norm that represents the number of non-zero elements, λ is a parameter that controls the sparsity of S, and $||\cdot||_2$ is the l_2-norm. The use of the l_0-norm cancels out the outliers in X, making the estimation more robust against outliers. However, this optimization (1) is NP-hard. To mitigate this issue, a convex relaxation has been proposed as follows:

$$\min_{L,S} ||L||_* + \lambda||S||_1 \quad \text{s.t. } ||X - L - S||_2^2 = 0, \tag{2}$$

where $||\cdot||_*$ is the nuclear norm and $||\cdot||_1$ is the l_1-norm. In general, the outlierness of each sample is obtained by adding $S \circ S$ along the feature dimension, where $\circ$ is the element-wise product.

Robust Deep Autoencoder: RDA [24] is a method that relaxes the linearity assumption of RPCA. RDA uses an AE instead of linear mapping. We denote the model parameters of an AE as θ and an output of the AE with a certain input and parameters θ as $f_{AE}(\cdot;\theta)$. Concretely, RDA aims to decompose X as $X = L_D + S$, where S is a sparse error matrix, the non-zero elements of which indicate reconstruction difficulty, and L_D is easily reconstructable data for AE. This is defined as the following l_1-relaxed optimization problem:

$$\min_{\theta,S} ||L_D - f_{AE}(L_D;\theta)||_2 + \lambda||S||_1 \quad \text{s.t. } X - L_D - S = 0. \tag{3}$$

An alternating optimization method for θ and S was proposed (see [24] for details) for optimization. Note that RDA is equivalent to AE when $\lambda = \inf$. In real applications, outliers are often *structured* [24], i.e., outliers are concentrated on a specific sample. For such cases, the use of grouped norm regularization instead of the l_1-norm in Eq. (3) has been proposed:

$$||S||_{2,1} = \sum_{j=1}^{D} ||\boldsymbol{s}_j||_2 = \sum_{j=1}^{D} (\sum_{i=1}^{N} |s_{ij}|^2)^{1/2}. \tag{4}$$

3 L0-norm Constrained Autoencoders

Although RDA can detect outliers even for nonlinear data, there are several concerns with RDA. First, owing to the NP-hardness, RDA uses the l_1-norm instead of the l_0-norm, which causes sensitivity to outliers. Second, the alternating optimization method of RDA does not include a theoretical analysis of convergence. In practice, it has been experimentally confirmed that the progress of training the RDA model may be unstable. To address these issues, we propose an unsupervised outlier detection method that can decompose data nonlinearly using AE under an l_0-norm constraint for the sparse matrix S. We prove that our algorithm always converges under a certain condition. For clarity, we first describe L0-AE for unstructured outliers and then extend it for structured outliers.

3.1 Formulation and Alternating Optimization Algorithm

Considering that all elements may contain some errors in real datasets, we decompose X into $\bar{L} = f_{AE}(X;\theta)$, a sparse error matrix S, and a small error matrix E as in Stable Principal Component Pursuit [25]:

$$X = f_{AE}(X;\theta) + S + E. \tag{5}$$

To train an AE that captures the features of X successfully, $||E||_2^2 = ||X - f_{AE}(X;\theta) - S||_2^2$ must be as small as possible. For optimization, we minimize E while adjusting the sparsity of S using the parameter $k \geq 0$ as follows:

$$\min_{\theta,S} ||X - f_{AE}(X;\theta) - S||_2^2 \quad \text{s.t. } ||S||_0 \leq k. \tag{6}$$

By solving (6), we can obtain a low-dimensional manifold that captures the nonlinear features of X and can completely avoid the influence of outliers.

In the following, we propose an alternating optimization algorithm for θ and S for the l_0-norm constrained optimization problem (6). We denote $X - f_{AE}(X;\theta)$ as $Z(\theta)$; then the objective function can be expressed as $||Z(\theta) - S||_2^2$. In the optimization phase of θ with S fixed, we employ a gradient-based method. With θ fixed, the optimal S is obtained in a closed form; it is the matrix that zeroes out the elements with the top-k largest absolute values in $Z(\theta)$, which can be written as follows:

$$s_{ij} = \begin{cases} z_{ij} \ (|z_{ij}| \geq c) \\ 0 \quad \text{(otherwise)}, \end{cases} \tag{7}$$

where c is the k-th largest value in $\{|z_{ij}| \mid 1 \leq i \leq N, 1 \leq j \leq D\}$.

We rewrite our proposed formulation (6) and alternating optimization method to be algorithmically concise as follows:

$$\min_{A,\theta} ||A \circ Z(\theta)||_2^2 \quad \text{s.t. } ||A||_0 \geq ND - k, \tag{8}$$

where $A \in \{0,1\}^{N \times D}$ is a binary-valued matrix. In the alternating optimization of Eq. (8), θ is optimized by gradient-based optimization and A is optimized by

$$a_{ij} = \begin{cases} 1 \ (|z_{ij}| < c) \\ 0 \ \text{(otherwise)}. \end{cases} \tag{9}$$

The procedure of our proposed optimization algorithm is as follows:

Input: $X \in \mathbb{R}^{N \times D}$, $k \in [0, N \times D]$ and $Epoch_{\max} \in \mathbb{N}$

Initialize $A \in \mathbb{R}^{N \times D}$ as a zero matrix, epoch counter $Epoch = 0$, and an autoencoder $f_{AE}(\cdot\,;\theta)$ with randomly initialized parameters.

Repeat the following *Epoch* times:
1. Obtain reconstruction error matrix Z: $Z = X - f_{AE}(X;\theta)$
2. Optimize A with θ fixed:
Get threshold $c = k$-th largest absolute value in Z and update A using Eq. (9)
3. Update θ with A fixed:
Minimize $||A \circ Z(\theta)||_2^2$ using gradient-based optimization

Return the elementwise outlierness $R \in \mathbb{R}^{N \times D}$ computed as follows:

$$R = (X - f_{AE}(X;\theta)) \circ (X - f_{AE}(X;\theta)). \tag{10}$$

In step 3, the number of iterations in each gradient-based optimization process affects the performance of L0-AE. In practice, L0-AE shows sufficient detection accuracy and convergence without iteration (see Sect. 5). In this case, the total computational cost of L0-AE is the sum of that the cost of normal AE and sorting to obtain the top-k error value.

3.2 Convergence Property

In this section, we prove that our alternating optimization algorithm always converges under the assumption that AE is trained appropriately by gradient-based optimization. Here, we denote the objective function $||A \circ Z(\theta)||_2^2$ as $K(A, \theta)$ and the variables A and θ at the t-th step of each alternating optimization phase as A^t and θ^t, respectively. Under this assumption, the convergence of the proposed alternating optimization method can be shown as follows:

Theorem 1. *Suppose* $K(A^t, \theta^t)$ *is updated to* $K(A^{t+1}, \theta^t)$ *using Eq. (9), and assume that* $K(A^{t+1}, \theta^t) \geq K(A^{t+1}, \theta^{t+1})$ *with gradient-based optimization. Then there exists a value* $a^\infty \geq 0$ *such that* $\lim_{t\to\infty} K(A^t, \theta^t) = a^\infty$.

Proof. By updating with Eq. (9), the obtained A^* minimizes Eq. (8) for any $Z(\theta)$. Hence, for any θ^t, we have $K(A^t, \theta^t) \geq K(A^{t+1}, \theta^t)$. Furthermore, $K(A^{t+1}, \theta^t) \geq K(A^{t+1}, \theta^{t+1})$ holds by assumption, which indicates $K(A^t, \theta^t) \geq K(A^{t+1}, \theta^{t+1})$. This implies that a sequence $\{K(A^t, \theta^t)\}$ is a monotonically non-increasing and non-negative sequence. Therefore, by applying the monotone convergence Theorem [2], there exists a value $a^\infty = \inf_t\{K(A^t, \theta^t)\} \geq 0$.

Remark. The assumption $K(A^{t+1}, \theta^t) \geq K(A^{t+1}, \theta^{t+1})$ holds when the learning rate of the AE model is sufficiently small. Although this assumption might not hold for a fixed learning rate in practice, L0-AE shows better convergence than RDA (see Sect. 5.5).

3.3 Algorithm for Structured Outliers

In what follows, we describe an alternating optimization algorithm for data with structured outliers. In order to detect structured outliers, Eq. (8) and (9) are, respectively, reformulated as follows:

$$\min_{\theta, \boldsymbol{a}} ||(\boldsymbol{a}_N \mathbf{1}_D^T) \circ (X - f_{AE}(X; \theta))||_2^2 \quad \text{s.t. } ||\boldsymbol{a}||_0 \geq N - k, \tag{11}$$

$$a_i = \begin{cases} 1 \ (\sum_{j=1}^D (z_{ij})^2 < c') \\ 0 \ (\text{otherwise}), \end{cases} \tag{12}$$

where the subscripts N and D represent the number of elements in the column vector and c' is the k-th largest value of the vector $\sum_{j=1}^D (z_{\cdot j})^2$. The sample-wise outlierness $\boldsymbol{r}'$ is calculated using the R defined by Eq. (10) as follows:

$$r'_i = \sum_{j=1}^D R_{i,j}. \tag{13}$$

L0-AE uses this version of the formulation and the alternating optimization method for outlier detection.

As with the update of A using Eq. (9), the update of $\boldsymbol{a}$ using Eq. (12) always minimizes the objective function (11) with θ fixed. The convergence of this algorithm using Eq. (12) is easily proved in a similar manner with Theorem 1.

Remark. The concept of our optimization methodology for structured outliers can be regarded as Least Trimmed Squares (LTS) [17], in which the sum of all squared residuals except the largest k squared residuals is minimized.

4 Related Work

Recently, highly accurate neural network-based anomaly detection methods, such as AE, Variational Autoencoder (VAE), or Generative Adversarial Network-based methods [1,18,26], have been proposed; however, they assume a different problem setting from ours, i.e., training data does not include anomalous data, and finding anomalies in test datasets is the target task. Therefore, these methods do not have a mechanism that excludes outliers during training. In [8], the equivalence of the global optimum of the VAE and RPCA is shown under the condition that a decoder has some kind of affinity; however, connections between VAE and RPCA are not shown for general nonlinear activation functions.

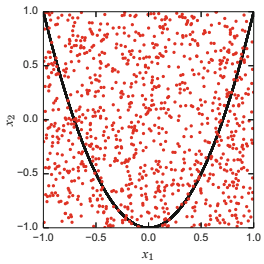


Fig. 1. Artificial dataset: black/red points are inliers/outliers. (Color figure online)

Table 2. Summary of the datasets

Dataset	Dims.	Samples	Outlier rate [%]
cardio	21	1,831	9.61
cover	10	286,048	0.96
kdd99_rev	118	121,597	20.00
mnist	100	7,603	9.21
musk	166	3,062	3.17
satellite	36	6,435	31.64
satimage-2	36	5,803	1.22
seismic	28	2,584	6.58
shuttle	9	49,097	7.15
smtp	3	95,156	0.03
thyroid	6	3,772	2.47
vowels	12	1,456	3.43

The Discriminative Reconstruction Autoencoder (DRAE) [20] has been proposed for unsupervised outlier removal. DRAE labels samples for which reconstruction errors exceed a threshold as "outliers" and omits such samples for learning. To appropriately determine the threshold, the loss function of DRAE has an additional term to separate the reconstruction errors of inliers and outliers. Because of this additional term, DRAE does not solve an l_0-norm constrained optimization problem, i.e., the learned manifold is affected by outlier values, which degrades the detection performance (see Sect. 5).

RandNet [7] has been proposed as a method to increase the robustness through an ensemble scheme. Although this ensemble may improve the robustness, it does not completely avoid the adverse effects of outliers, because each

AE uses an non-regularized objective function. Deep Structured Energy-Based Models [21], a robust AE-based method that combines energy-based models and non-regularized AE, has the same drawback. In [11], a method that simply combines AE and LTS was proposed; however, no theoretical analysis for the combined effects of AE and LTS was presented.

5 Experimental Results

5.1 Experimental Settings

Datasets. We employed both artificial and real datasets. Figure 1 illustrates the artificial data. We sampled 9,000 inlier samples $(x, 2x^2 - 1) \in \mathbb{R}^2$ where $x \in [-1, 1]$ was sampled uniformly. Further, we sampled 1,000 outliers uniformly from $[-1, 1] \times [-1, 1]$. As real datasets, we used 11 datasets from Outlier Detection DataSets (ODDS) [16], which are commonly used as benchmarks for outlier detection methods. In addition, we also used the "kdd99_rev" dataset introduced in [26]. Table 2 summarizes the 12 datasets. Before the experiments, we normalize the values of the datasets by dimension into the range of -1 to 1.

Evaluation Method. Following the evaluation methods in [7,14,15], we compared AUCs of the outlier detection accuracy. Evaluation was performed as follows: (1) all samples (whether inlier or outlier) were used for the training; (2) the outlierness of each sample was calculated after training; and (3) AUCs were calculated by using outlierness and inlier/outlier labels. Note that we need not specify the detection threshold in this evaluation scheme.

5.2 Methods and Configurations

Robust PCA (RPCA). We utilize the RPCA implemented in [9] as a baseline linear method. We set tol = 1E-05 that is used to determine the convergence.

Normal Autoencoders (N-AE). We implemented N-AE with a loss function $||X - f_{AE}(X; \theta)||_2^2$ as a baseline non-regularized detection method. For every AE-based method below, we used common network settings. We used three fully connected hidden layers (with a total of five layers), in which the number of neurons was $ceil([D, D^{\frac{1}{2}}, D^{\frac{1}{4}}, D^{\frac{1}{2}}, D])$ from the input to the output unless otherwise noted. These were connected as (input layer) - linear - relu - (hidden layer1) - linear - (hidden layer2) - linear - relu - (hidden layer3) - linear - (output layer). We set the mini-batch size to $N/50$ and applied Adam [13] ($\alpha = 0.001$) for optimization with $Epoch_{\max} = 400$. To prevent undue advantages to our method (L0-AE) and the other AE-based methods, we searched this architecture by maximizing the average AUC of N-AE.

Robust Deep Autoencoders (RDA). We implemented the RDA [23] with the grouped norm version of Eq. (3). We use Eq. (13) to calculate the sample-wise outlierness of RDA. To make the number of loops equal to those of the other AE-based methods, the parameter *inner_iteration*, which is the number of iterations required to optimize AE during one execution of l_1-norm optimization, is set to 1. We set λ as 0.00005.

RDA-Stbl. This baseline is used to confirm the effect of the l_0-norm constraint of L0-AE. RDA-Stbl minimizes the objective function $||L_D - f_{AE}(X;\theta)||_2 + \lambda||S^T||_{2,1}$ such that $X - L_D - S = 0$, with respect to S and θ. This model can be regarded as a relaxed version of L0-AE. We set λ as 0.0005.

L0-norm Constrained Autoencoders (L0-AE). We use L0-AE for structured outliers (described in Sect. 3.3). The sample-wise outlierness of L0-AE is calculated using Eq. (13). We do not iterate for updating the parameters of an AE at each gradient-based optimization step. Instead of k, we use $C_p = k/N$ $(0 \leq C_p \leq 1)$, which is normalized by the number of samples, and set C_p as 0.3. This type of normalized parameter is often used in other methods such as the One-Class Support Vector Machine (OC-SVM) [19] and Isolation Forest (IForest) [15]. Note that L0-AE is equivalent to N-AE when $C_p = 0$.

Variational Autoencoder (VAE). We adopted VAE for our problem setting. The outlierness is computed using reconstruction probability [1]. Note that the number of output dimensions of hidden layer1 and layer3 is twice that of the other AE-based methods.

Discriminative Reconstruction Autoencoder (DRAE). We set λ as 0.1, which determines the weight of the term in the objective function for separating the inlier and outlier reconstruction errors (see Sect. 4).

We used Chainer (ver. 1.21.0) [6] for implementation of the AE-based methods above. In addition, we apply the following three conventional methods for a comparison of detection accuracy against real benchmark datasets.

One-Class Support Vector Machine. We use the OC-SVM implemented in scikit-learn and set *kernel* = 'rbf'.

Local Outlier Factor (LOF) [3]**.** We use the LOF implemented in scikit-learn and set "k" for the k-nearest neighbors to 100.

Isolation Forest. We used the IForest from a Python library *pyod* [22] with "*n-estimators*" (the number of trees) set to 100.

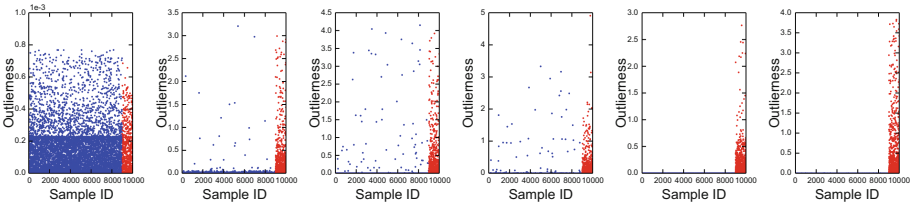
We tuned the above-mentioned parameters that control the robustness against noise to achieve high AUC on average over all real datasets; for the other parameters, we used the recommended (default) values unless otherwise noted.

5.3 Robustness for Corrupted Data

We evaluate the robustness against outliers of L0-AE and the baseline reconstruction-based methods using the artificial data. We compare the average AUC, as well as the average outlierness of inlier samples O^i_{avg}, average outlierness of outlier samples O^o_{avg}, and the ratio O^o_{avg}/O^i_{avg} (a higher value implies that less outliers are close to a low-dimensional manifold). In this experiment, because $D = 2$, we could not set the number of neurons and parameters as mentioned in Sect. 5.2; instead, for N-AE to achieve a high AUC, we used $[2, 100, 1, 100, 2]$, which are empirically obtained. For RDA and RDA-Stbl, we used $\lambda = 0.00001$; for DRAE, $\lambda = 0.1$; and for L0-AE, $C_p = 0.2$. These are chosen based on the AUC values.

Table 3. Average measurements from L0-AE and other reconstruction-based methods

Methods	RPCA	NAE	RDA	RDA-Stbl	VAE	DRAE	L0-AE
AUC[%]	39.9	93.9	97.8	96.5	79.4	97.4	**99.8**
O^i_{avg}	1.30E−04	3.90E−03	7.12E−03	7.68E−03	–	6.57E−04	1.27E−05
O^o_{avg}	9.80E−05	1.60E−01	3.16E−01	1.70E−01	–	1.63E−01	4.56E−01
O^o_{avg}/O^i_{avg}	0.75	42.09	44.41	22.14	–	247.60	**35965.54**



(a) RPCA (b) N-AE (c) RDA (d) RDA-Stbl (e) DRAE (f) L0-AE

Fig. 2. Distributions of outlierness for each method in the first run: sample ID of inliers and outliers are 1–9,000 and 9,001–10,000, respectively.

Table 3 reports the average of these measurements over 20 trials with different initial network weights, and Fig. 2 shows the distributions of outlierness of each method except RPCA in the first run. As VAE uses the probability as outlierness, only the AUC is included in Table 3. These results show that L0-AE outperforms the other methods in terms of AUC and the distribution of outlierness between inliers and outliers (i.e., sparsity of the error matrix).

RPCA performs significantly poorer than the other methods because of its linearity. Among the AE-based methods, L0-AE shows the best performance.

Table 4. Comparison of AUCs[%] with standard deviation

Methods	cardio	cover	kdd99.	mnist	musk	satel.	satim.	seismic	shuttle	smtp	thyroid	vowels	Avg. AUC	Avg. rank	Avg. time
RPCA	93.8±0.0	**95.9**±0.0	25.7±0.0	82.3±0.0	97.0±0.0	61.9±0.0	98.0±0.0	68.0±0.0	97.7±0.0	80.0±0.0	93.1±0.0	88.8±0.0	81.86	4.92	3.4
N-AE	80.7±6.9	87.9±9.9	14.5±4.1	83.7±1.0	67.7±12.4	64.6±1.1	78.1±2.3	70.8±2.5	77.0±16.9	75.9±6.6	89.1±2.2	90.1±2.5	73.34	6.75	57.6
RDA	83.4±6.9	88.6±8.0	13.3±2.6	82.7±3.8	70.9±15.5	63.2±2.3	77.7±8.9	**72.1**±2.1	80.1±17.9	77.9±6.7	89.9±5.3	90.8±3.3	74.21	6.08	66.4
RDA-Stbl	86.0±8.1	87.1±11.6	12.7±3.4	82.6±1.3	68.0±11.2	63.7±1.6	77.6±3.0	71.8±1.3	74.6±18.2	74.7±5.2	90.0±3.0	89.1±2.5	73.13	7.25	77.5
VAE	72.1±4.4	81.5±4.6	55.1±23.3	81.7±14.7	34.3±33.8	67.7±5.5	94.1±5.0	69.5±3.1	98.3±3.0	87.1±4.0	97.0±2.1	83.9±3.3	76.86	5.92	139.8
DRAE	89.5±6.9	91.5±1.1	85.4±25.8	84.4±2.6	**100.0**±0.0	65.8±3.2	90.1±12.6	58.6±10.8	78.9±14.5	83.5±6.2	94.2±2.5	90.1±3.4	84.35	4.50	2869.3
L0-AE	**94.0**±3.7	87.2±6.4	**96.2**±1.2	**86.2**±2.2	**100.0**±0.0	**75.6**±3.1	**99.4**±2.4	66.9±3.0	97.7±6.1	89.0±5.2	93.8±3.6	90.9±2.9	**89.74**	**2.92**	74.5
OC-SVM	93.0±0.0	91.8±0.0	81.5±0.0	82.0±0.0	93.1±0.0	59.9±0.0	98.0±0.0	59.3±0.0	98.3±0.0	76.9±0.0	85.0±0.0	57.3±0.0	81.35	6.00	668.6
LOF	85.1±0.0	60.1±0.0	36.3±0.0	80.3±0.0	84.0±0.0	56.7±0.0	96.3±0.0	59.2±0.0	56.8±0.0	87.6±0.0	96.3±0.0	**93.4**±0.0	74.33	6.58	46.6
IForest	92.2±1.3	87.2±3.2	78.1±2.9	79.8±2.0	99.9±0.1	70.5±1.8	99.3±0.1	67.3±0.6	**99.7**±0.1	**90.7**±0.6	**97.7**±0.3	75.4±2.4	86.49	4.08	2.3

In L0-AE, we can see that the learned manifold is almost entirely composed of inliers. Therefore, it can be confirmed that the l_0-norm constraint of L0-AE functions as intended, and L0-AE can learn by almost completely eliminating the influence of the corrupted samples while capturing nonlinear features. In contrast, the performances of the other AE-based methods are inferior to that of L0-AE because the other methods cannot completely exclude the influence of outliers. VAE is less accurate than the other AE-based methods; it is considered that VAE is unable to demonstrate robustness owing to the non-affinity of the decoder. For DRAE, the reconstruction errors of inliers and outliers are relatively well separated, but DRAE is more strongly affected by outliers than L0-AE because the DRAE objective function depends on how large outliers are, while the L0-AE objective function does not.

5.4 Evaluation of Accuracy and Parameter Sensitivity

We compare the detection accuracy for the real datasets. The AUC values are averaged over 50 trials with different random seeds. Table 4 presents the average AUCs for each dataset; *Avg. AUC*, *Avg. rank*, and *Avg. time* refer to the average AUC, the average rank over the datasets, and the average run-time, respectively.

L0-AE demonstrates the highest average AUC and average rank. Among the reconstruction-based methods, L0-AE showed the highest AUCs for 8 out of 12 datasets. Especially on kdd99_rev, the AUC of L0-AE is considerably higher than those of the other AE-based methods. Because kdd99_rev has a high rate of outliers and they are distributed close to each other, the methods with l_1-norm regularization and no regularization cannot avoid reconstructing the outliers, whereas L0-AE can almost completely avoid reconstruction because of its l_0-norm constraint. Furthermore, we observed that the AUCs of RDA and RDA-Stbl are nearly equal. This shows the importance of the l_0-norm constraint. L0-AE outperforms DRAE on average; it is considered that L0-AE selectively reconstructs only the inliers, while DRAE reconstructs inliers and reduces the variance of each label, allowing outliers to affect manifolds. In addition, the computational cost of DRAE is higher than that of L0-AE, owing to the calculation of the threshold. For VAE, the training was unstable for some datasets. One possible reason is that VAE involves random sampling in the *reparametrization trick* which increases the randomness of the results under these experimental settings. In contrast, among the AE-based methods, L0-AE showed stable results. RPCA results are relatively good in some datasets, suggesting that these datasets have linear features and l_0-norm regularization works; L0-AE shows good performance by capturing nonlinear features even for the other datasets. The reason why RPCA outperforms some AE-based methods on average is that RPCA can automatically detect the rank of the inlier, while the AE-based methods have a fixed latent dimension (there is no known method for obtaining an appropriate latent dimension in an unsupervised setting).

Next, we evaluate the parameter sensitivity of L0-AE using real datasets. Fig. 3 shows the AUCs with different C_p values for L0-AE (averaged over 50 trials). Overall, the maximum AUC values occur at C_p values moderately greater

than the true outlier rates. If C_p is greater than the true outlier rate, outliers are safely detected as outliers; in contrast, there are inliers that are not trained to be reconstructed at an epoch. However, such inliers are also trained to be well reconstructed because inliers are likely to be distributed close to each other. If C_p is less than the true outlier rate, the detection accuracy is basically better than in the case of N-AE ($C_p = 0$) because some outliers are not reconstructed. For kdd99_rev, owing to the distribution of outliers as mentioned above, the outliers are unexpectedly detected as inliers when C_p is small; for large C_p values, such outliers are safely detected as outliers. Therefore, the change in AUC against C_p is large. The development of an automatic optimal C_p search method under the l_0-norm constraint without the ground truth labels is an important future work.

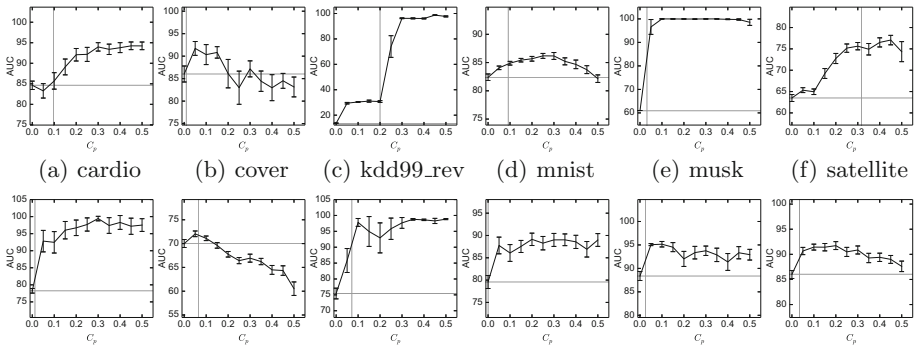


Fig. 3. Parameter sensitivity of L0-AE for real datasets; each error bar represents a 95% confidence interval, gray vertical lines indicate the true outlier rates of each dataset, and gray horizontal lines indicate the AUCs of normal AE ($C_p = 0$) for each dataset.

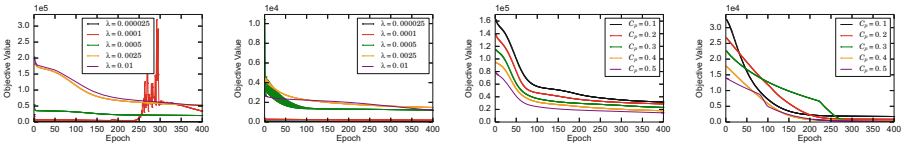
5.5 Evaluation of Convergence

We compare the convergence of L0-AE with that of RDA. Here, we do not use mini-batch training to remove the effect of randomness. Table 5 presents the sum of the number of epochs in which the value of the objective function has increased over the previous epoch during 20 trials (96,000 epochs in total). The results of N-AE are also included for reference. In addition, Fig. 4 shows two transition examples of the values of the objective functions of RDA and L0-AE. Among them, Fig. 4(d) shows the result of the only trial in which the objective function increased in L0-AE; the epochs in which the objective function increased were 294 to 302 when $C_p = 0.3$, with an average increase of 0.23, which is considerably less than the value of the objective function. In Table 5 and Fig. 4, we observe that L0-AE shows good convergence regardless of the parameter C_p, unlike RDA. This empirically demonstrates the validity of Theorem 1, which

states that our alternating optimization algorithm converges when the gradient-based optimization behaves ideally. For RDA, when λ is small, the value of the objective function is unstable, but when λ is large, the characteristic of RDA approaches N-AE; therefore, the stability improves. We observe that, with N-AE, the values of objective function do not increase, which implies that our gradient-based optimization basically satisfies the assumption in Theorem 1.

Table 5. Number of epochs in which the objective function value has increased over the previous epoch (summed over all datasets)

N-AE	L0-AE $C_p =$					RDA $\lambda =$				
	0.1	0.2	0.3	0.4	0.5	0.000025	0.0001	0.0005	0.0025	0.01
0	0	0	9	0	0	3152	2031	1323	777	15



(a) RDA: musk (b) RDA: vowels (c) L0-AE: musk (d) L0-AE: shut.

Fig. 4. Examples of the transition of the values of the objective functions from RDA and L0-AE

6 Conclusion

In this paper, we proposed L0-norm Constrained Autoencoders (L0-AE) for unsupervised outlier detection. L0-AE decomposes data nonlinearly into a low-rank matrix and a sparse error matrix under the l_0-norm constraint. We proposed an efficient alternating optimization algorithm for training L0-AE and proved that this algorithm converges under a mild condition. We conducted extensive experiments with real and artificial data and confirmed that L0-AE is highly robust to outliers. We also confirmed that this high robustness leads to higher outlier detection accuracy than those of existing outlier detection methods.

References

1. An, J., Cho, S.: Variational autoencoder based anomaly detection using reconstruction probability. Technical report 1, Special Lecture on IE (2015)
2. Bibby, J.: Axiomatisations of the average and a further generalisation of monotonic sequences. Glasg. Math. J. **15**(1), 63–65 (1974)
3. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: LOF: identifying density-based local outliers. In: ACM SIGMOD Record, vol. 29, pp. 93–104. ACM (2000)

4. Bui, T.D., Quach, K.G., Duong, C.N., Luu, K.: LP norm relaxation approach for large scale data analysis: a review. In: ICIAR 2018, pp. 285–292 (2018)
5. Candès, E.J., Li, X., Ma, Y., Wright, J.: Robust principal component analysis? J. ACM **58**(3), 1–37 (2011)
6. Chainer: a flexible framework for neural networks (2019). https://chainer.org/
7. Chen, J., Sathe, S., Aggarwal, C., Turaga, D.: Outlier detection with autoencoder ensembles. In: ICDM 2017, pp. 90–98. SIAM (2017)
8. Dai, B., Wang, Y., Aston, J., Hua, G., Wipf, D.: Connections with robust pca and the role of emergent sparsity in variational autoencoder models. J. Mach. Learn. Res. **19**(1), 1573–1614 (2018)
9. Ganguli, D.: GitHub - dganguli/robust-PCA: A simple python implementation of R-PCA (2019). https://github.com/dganguli/robust-pca
10. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006)
11. Ishii, Y., Takanashi, M.: Low-cost unsupervised outlier detection by autoencoders with robust estimation. J. Inf. Process. **27**, 335–339 (2019)
12. Jolliffe, I.: Principal Component Analysis. Springer, Heidelberg (2011)
13. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
14. Kriegel, H.P., Kröger, P., Schubert, E., Zimek, A.: LoOP: local outlier probabilities. In: CIKM 2009, pp. 1649–1652. ACM (2009)
15. Liu, F.T., Ting, K.M., Zhou, Z.H.: Isolation forest. In: ICDM 2008, pp. 413–422. IEEE (2008)
16. Rayana, S.: Odds library (2019). http://odds.cs.stonybrook.edu
17. Ruppert, D., Carroll, R.J.: Trimmed least squares estimation in the linear model. J. Am. Stat. Assoc. **75**(372), 828–838 (1980)
18. Schlegl, T., Seeböck, P., Waldstein, S.M., Schmidt, U., Langs, G.: Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: Niethammer, M., et al. (eds.) IPMI 2017, vol. 10265, pp. 146–157. Springer, Heidelberg (2017). https://doi.org/10.1007/978-3-319-59050-9_12
19. Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C.: Estimating the support of a high-dimensional distribution. Neural Comput. **13**(7), 1443–1471 (2001)
20. Xia, Y., Cao, X., Wen, F., Hua, G., Sun, J.: Learning discriminative reconstructions for unsupervised outlier removal. In: IEEE ICCV 2015, pp. 1511–1519 (2015)
21. Zhai, S., Cheng, Y., Lu, W., Zhang, Z.: Deep structured energy based models for anomaly detection. arXiv preprint arXiv:1605.07717 (2016)
22. Zhao, Y.: PyOD (2019). http://pyod.readthedocs.io/en/latest/
23. Zhou, C.: GitHub - zc8340311/robustautoencoder: a combination of autoencoder and robust PCA (2019). https://github.com/zc8340311/RobustAutoencoder
24. Zhou, C., Paffenroth, R.C.: Anomaly detection with robust deep autoencoders. In: KDD 2017, pp. 665–674 (2017)
25. Zhou, Z., Li, X., Wright, J., Candes, E., Ma, Y.: Stable principal component pursuit. In: ISIT 2010, pp. 1518–1522. IEEE (2010)
26. Zong, B., et al.: Deep autoencoding Gaussian mixture model for unsupervised anomaly detection. In: ICLR 2018 (2018)